REMARKS

Applicant respectfully requests reconsideration and allowance of the subject application. Claims 1, 15, 19, and 25 are amended. Claims 1-41 are pending in this application.

Claim Objections

Claim 25 stands objected to due to informalities. As part of this Response, the informalities noted in the November 3, 2004 Office Action have been corrected. Applicant respectfully requests that the objection to the claims be withdrawn.

35 U.S.C. § 101

Claims 1-12 and 19-24 stand rejected under 35 U.S.C. §101. As part of this Response, independent claims 1 and 19 have been amended to clarify that the claims are directed to statutory subject matter. Applicant respectfully submits that claims 1-12 and 19-24 comply with 35 U.S.C. §101.

Applicant respectfully requests that the §101 rejections be withdrawn.

35 U.S.C. § 103

Claims 1-3, 4, 6-17, 19-21, 23-27, 29-38, and 40-41 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,138,271 to Keeley (hereinafter "Keeley") in view of U.S. Patent No. 5,325,533 to McInerney et al. (hereinafter "McInerney"). Applicant respectfully submits that claims 1-3, 4, 6-

17, 19-21, 23-27, 29-38, and 40-41 are not obvious over Keeley in view of McInerney.

Keeley is directed to a computer operating system adapted for use in a limited memory computer that may be embedded in other products (see, col. 1, lines 6-8). More specifically, as discussed in the Abstract of Keeley, Keeley is directed to a software development system including a modular operating system program having a plurality of modules each providing an operating system operation that may be called by an application program. The operating system operations permit application programs to communicate with particular computer hardware or other executing programs. A scanner program reviews the application program for calls to operating system operations to produce an OS module list of such application calls; and a selective compiler program receiving the OS module list prepares an operating system comprised of only those modules of the modular operating system necessary to perform the application calls of the OS module list that may be loaded into the embedded computer.

McInerney is directed to a human oriented object programming system (HOOPS) which provides an interactive and dynamic environment for computer program building (see, col. 1, lines 9-11). Program building is made possible by the interaction of an incremental program model, called a project, and three major functionalities (see, col. 3, lines 4-7). A program is modeled as semantic units called components, each component representing a single compilable language element such as a class or a function (see, col. 3, lines 7-8 and 15-17). The three major functionalities are the database, the compiler and the build mechanism (see, col. 3, lines 25-26). The database persistently stores and retrieves the components

and their properties (see, col. 3, lines 26-28). The compiler, along with compiling the source code of a property, is responsible for calculating the dependencies associated with a component (see, col. 3, lines 28-30). The build mechanism uses properties of components along with the compiler generated dependencies to correctly and efficiently sequence the compilation of components during a build process (see, col. 3, lines 30-34).

In contrast, claim 1 recites in part:

> creating a software development kit object (SDK object) for at least some of a plurality of development files in a source operating system that includes development files and components;
> identifying features of the source operating system to be included in a modularized system that is a subset of the source operating system;
> tracing dependencies in a dependency model correlating to the source operating system that uses the SDK objects to identify SDK objects corresponding to development files that are required to support the identified features;
> selecting the development files that correspond to the identified SDK objects; and
> exporting the selected development files to a software development kit (SDK) that supports development of applications for use with the modularized system.

Applicant respectfully submits that Keeley in view of McInerney does not disclose or suggest the method of claim 1.

Keeley is cited in the November 3, 2004 Office Action as disclosing the creating, identifying, selecting, and exporting of claim 1 (See, November 3, 2004 Office Action at ¶ 8, pp. 3-4). However, there is no disclosure or suggestion in Keeley of creating a software development kit object for at least some of a plurality of development files in a source operating system that includes development files and components as recited in claim 1. Keeley discusses a

software development system including a modular operating system program having a plurality of modules (see, col. 2, lines 62-65), and a selective compiler program that prepares an operating system comprised of only modules of the modular operating system necessary to perform the calls to operating system operations in an application program (see, col. 3, lines 1-8). Thus, Keeley discusses a software development system that prepares an operating system comprised of only some of the modules of the modular operating system, not creating a software development kit object as recited in claim 1.

Furthermore, Keeley discloses that development of an application program may be done on a standard desk-top computer system using the entire modular operating system installed or an equivalent non-modular operating system (see, col. 9, lines 9-13). In fact, one of the objects of Keeley is to permit programmers developing applications for embedded computers to develop their application programs using a full-featured operating system that supports high level languages and development tools (see, col. 3, lines 14-18). In contrast, claim 1 recites exporting the selected development files to a software development kit (SDK) that supports development of applications for use with a modularized system, the selected development files corresponding to identified SDK objects that in turn correspond to development files that are required to support features of the source operating system identified to be included in the modularized system that is a subset of the source operating system. If the goal of Keeley is to permit programmers to develop their applications using a full-featured operating system, it would be nonsensical to perform any of the identifying, selecting or exporting as recited in claim 1 – Keeley is using a full-featured operating system for

17

development so there would be no need to differentiate between development files that are required to support the identified features and those that are not required.

With respect to McInerney, McInerney is not cited as curing these deficiencies of Keeley, and Applicant respectfully submits that McInerney does not cure these deficiencies of Keeley.

Furthermore, McInerney is cited in the November 3, 2004 Office Action as disclosing the tracing dependencies in a dependency model correlating to the source operating system that uses the SDK objects to identify SDK objects corresponding to development files that are required to support the identified features of claim 1 (See, November 3, 2004 Office Action at ¶ 8, p. 4). As discussed above, McInerney is directed to a human oriented object programming system (HOOPS) which provides an interactive and dynamic environment for computer program building. There is no discussion or mention in McInerney of software development kit objects or tracing dependencies to identify software development kit objects as recited in claim 1. Absent such a discussion or mention, Applicant respectfully submits that McInerney cannot disclose or suggest tracing dependencies in a dependency model correlating to the source operating system that uses the SDK objects to identify SDK objects corresponding to development files that are required to support the identified features as recited in claim 1. In the November 3, 2004 Office Action it is acknowledged that Keeley does not disclose the tracing of claim 1 (See, November 3, 2004 Office Action at ¶ 8, p. 4). As such, Applicant respectfully submits that Keeley in view of McInerney does not disclose or suggest the tracing of claim 1.

For at least these reasons, Applicant respectfully submits that claim 1 is allowable over Keeley in view of McInerney.

With respect to claim 8, claim 8 depends from claim 1 and Applicant respectfully submits that claim 8 is allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 1. Furthermore, claim 8 recites:

> The method as recited in claim 1, wherein creating the SDK objects further comprises:
> naming a data object having a type that identifies the data object as being an SDK object;
> including at least one reference in a first SDK object, the reference pointing to a second SDK object that is required by the first SDK object to function properly; and
> repeating the previous steps for each development file to be exposed in the SDK.

Applicant respectfully submits that no software development kit object (SDK object) is disclosed in Keeley, much less naming a data object having a type that identifies the data object as being an SDK object as recited in claim 1.

In the November 3, 2004 Office Action, Keeley at col. 5, lines 6-7 is cited as disclosing the naming of claim 8 (see, November 3, 2004 Office Action at ¶8, p. 9). Applicant respectfully disagrees and submits that this cited portion of Keeley does not disclose the naming of claim 8. The sentence including this cited portion of Keeley reads:

> The application program calls the operating system operations via an application programmer's interface ("API") 28 which provides a standard set of operation names 40 that are linked to particular modules 44 of the operating system holding corresponding operations, each operation being a routine for performing the desired operation (e.g., a disk read, etc.).

Thus, it can be seen that Keeley is discussing an application programmer's interface and operation names provided by such an interface. There is no reference at all to a software development kit object in this portion of Keeley.

Without even so much as a mention of a software development kit object, Applicant respectfully submits that Keeley cannot disclose or suggest naming a data object having a type that identifies the data object as being an SDK object as recited in claim 8.

For at least these reasons, Applicant respectfully submits that claim 8 is allowable over Keeley in view of McInerney.

Given that claims 2-4, 6, 7, and 9-12 depend from claim 1, Applicant respectfully submits that claims 2-4, 6, 7, and 9-12 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 1.

With respect to claim 13, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the selecting and exporting of claim 13. For at least these reasons, Applicant respectfully submits that claim 13 is allowable over Keeley in view of McInerney.

Given that claims 14-17 depend from claim 13, Applicant respectfully submits that claims 14-17 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 13.

With respect to claim 19, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the identifying, selecting, and exporting of claim 19. For at

least these reasons, Applicant respectfully submits that claim 19 is allowable over Keeley in view of McInerney.

Given that claims 20-21 and 23-34 depend from claim 19, Applicant respectfully submits that claims 20-21 and 23-34 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 19.

With respect to claim 25, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the selecting and filtering of claim 25. For at least these reasons, Applicant respectfully submits that claim 25 is allowable over Keeley in view of McInerney.

With respect to claim 26, claim 26 depends from claim 25 and Applicant respectfully submits that claim 26 is allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 25. Furthermore, claim 26 recites:

> The one or more computer-readable media as recited in claim 25, wherein the filtering further comprises:
> for each selected SDK object, searching for a label in the master SDK header file that is the same name as the SDK object;
> if the label is found, enabling a section of code associated with the label; and
> wherein the enabling the section of code associated with the label enables the appropriate development file associated with the SDK object to be included in the SDK.

Applicant respectfully submits that no SDK object is disclosed in Keeley, much less for each selected SDK object, searching for a label in the master SDK header file that is the same name as the SDK object as recited in claim 26.

Applicant respectfully submits that there is no discussion or mention of a master SDK header file in Keeley or McInerney, and that there is no portion of either Keeley or McInerney cited in the November 3, 2004 Office Action as disclosing a master SDK header file. Without such a discussion or mention, Applicant respectfully submits that Keeley and McInerney cannot disclose or suggest searching for a label in a master SDK header file as recited in claim 26. Furthermore, similar to the discussion above regarding claim 8, there is no discussion or mention in Keeley or McInerney of a name of an SDK object as recited in claim 26. Without such a discussion or mention, Applicant respectfully submits that Keeley and McInerney cannot disclose or suggest searching for a label in the master SDK header file that is the same name as the SDK object as recited in claim 26.

For at least these reasons, Applicant respectfully submits that claim 26 is allowable over Keeley in view of McInerney.

With respect to claim 27, claim 27 depends from claim 25 and Applicant respectfully submits that claim 27 is allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 25. Furthermore, Applicant respectfully submits that, similar to the discussion of claim 26 above, Keeley in view of McInerney does not disclose or suggest the executing and determining of claim 27. For at least these reasons, Applicant respectfully submits that claim 27 is allowable over Keeley in view of McInerney.

Given that claims 29-33 depend from claim 25, Applicant respectfully submits that claims 29-33 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 25.

With respect to claim 34, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the SDK object generator, the feature identification module, the dependency tracer, and the export module of claim 34. For at least these reasons, Applicant respectfully submits that claim 34 is allowable over Keeley in view of McInerney.

Given that claims 35-38 and 40-41 depend from claim 34, Applicant respectfully submits that claims 35-38 and 40-41 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 34.

Claims 5, 18, 22, 28, and 39 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Keeley in view of U.S. Patent No. 5,901,319 to Hirst (hereinafter "Hirst"). It appears from the November 3, 2004 Office Action that McInerney is also relied on in the rejection of claims 5, 18, 22, 28, and 39 (see, November 3, 2004 Office Action at ¶ 9, p. 16). Applicant respectfully submits that claims 5, 18, 22, 28, and 39 are not obvious over Keeley in view of Hirst.

Claims 5, 18, 22, 28, and 39 depend from claims 1, 13, 19, 25, and 34, respectively. Applicant respectfully submits that claims 5, 18, 22, 28, and 39 are allowable over Keeley in view of McInerney at least because of their dependency on claims 1, 13, 19, 25, and 34, respectively. Applicant further submits that Hirst is not cited as curing, and does not cure, the deficiencies of Keeley in view of McInerney discussed above. For at least these reasons, Applicant respectfully submits that claims 5, 18, 22, 28, and 39 are allowable over Keeley in view of Hirst, as well as Keeley in view of McInerney and Hirst.

Applicant respectfully requests that the §103 rejections be withdrawn.

Conclusion

Claims 1-41 are in condition for allowance. Applicant respectfully requests reconsideration and issuance of the subject application. Should any matter in this case remain unresolved, the undersigned attorney respectfully requests a telephone conference with the Examiner to resolve any such outstanding matter.

Respectfully Submitted,

Date: 4/28/05

By: _____
Allan T. Sponseller
Reg. No. 38,318
(509) 324-9256

24   Application No. 10/028,739